中山大學
SUN YAT-SEN UNIVERSITY

CIKM2019

CamDrop: A New Explanation of Dropout and A Guided Regularization Method for Deep Neural Networks

Hongjun Wang, Guangrun Wang, Guanbin Li, Liang Lin

Sun Yat-sen University

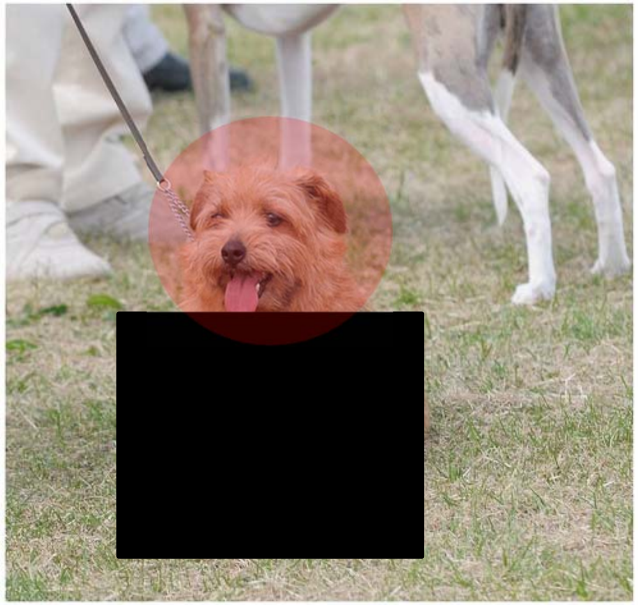
Outline

- **Introduction**
- Method
- Experiment
- Conclusion

中山大學 SUN YAT-SEN UNIVERSITY

Introduction

- Take 5 seconds to guess which category these images belong to.
  - It’s easy, right?
  - Because you can recognize them by **their distinctive features**



SUN YAT-SEN UNIVERSITY

Introduction

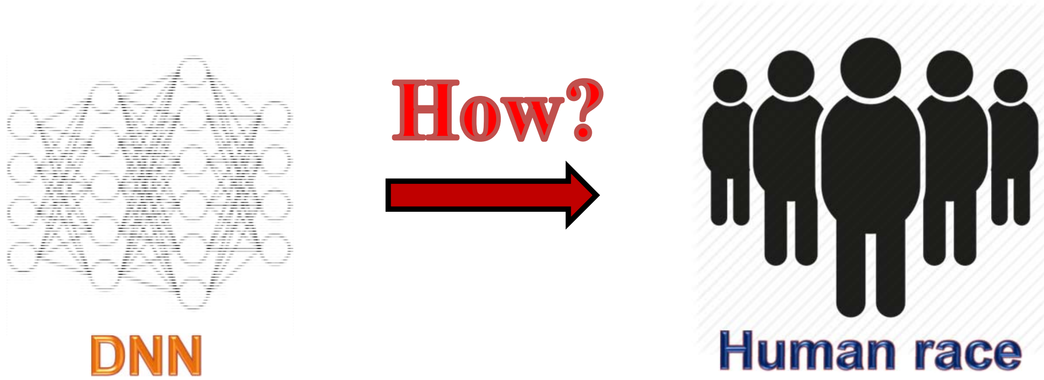
- What about these two?
  - Is it a little hard?
  - But you can still speculate their possible categories by **other neglected parts**

Cat or Dog

Tractor

The gap

- Human race can speculate object categories even though the dominant part is occluded



  - Representations extracted by a robust DNN can **represent more reasonable high-level semantics or detailed spatial information**
  - A robust DNN should be **discouraged from "outsmarting" itself by cutting corners**, as clinging to the most obvious point may be one-sided or false positive, which **also known as "overfitting"**

SUN YAT-SEN UNIVERSITY

Motivation

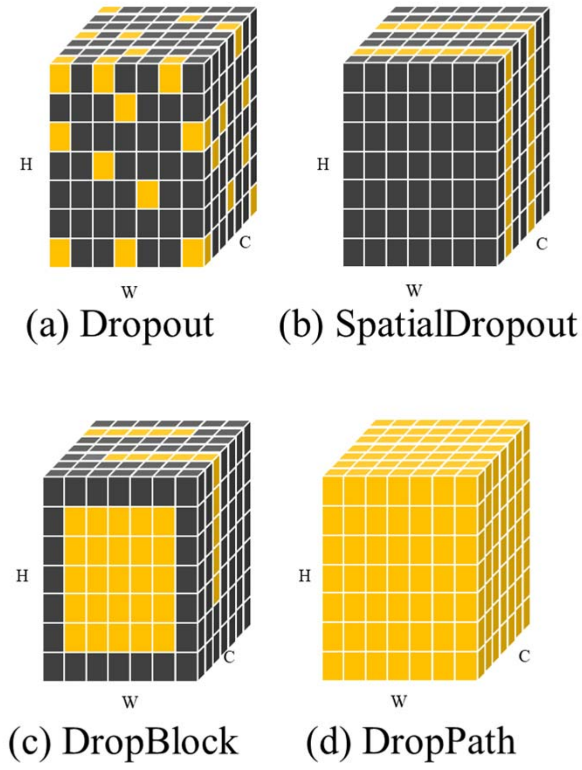
- Existing Dropout methods
  - Recent structured methods are better than others
  - But still lack of semantic guidance

$$\widetilde{X}^l = \mathbf{M}^l \odot X^l/\gamma$$

$$\mathbf{M}^l_{c,h,w} \sim \mathrm{Ber}(\gamma), \qquad \gamma \in [0, 1]$$

γ: the retaining rate

$\odot$: the element-wise product of tensors



(a) Dropout (b) SpatialDropout

(c) DropBlock (d) DropPath

Motivation

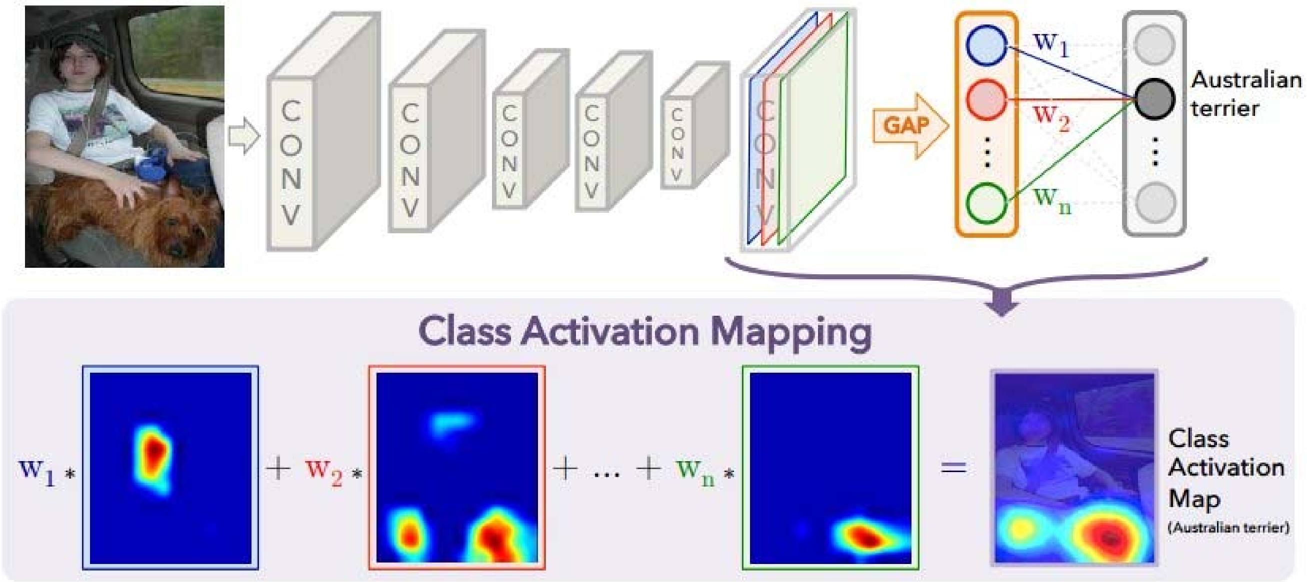
- Main idea
  - Utilize Class Activation Map (CAM) as the semantic guidance to ***hides discriminative object parts*** during training
  - Force the network to proactively ***explore other neglected parts autonomously*** instead of relying on external data

Outline

- Introduction
- **Method**
- Experiment
- Conclusion

Method

- Class Activation Map (CAM)
  - You can see that the CNN is triggered by different semantic regions of the image
  - CAM indicates **the discriminative image regions used by the CNN to identify the particular category**



B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning Deep Features for Discriminative Localization.

Method

(0) Generating **class activation map** $J^{k'} \in \mathbb{R}^{1\times H'\times W'}$:

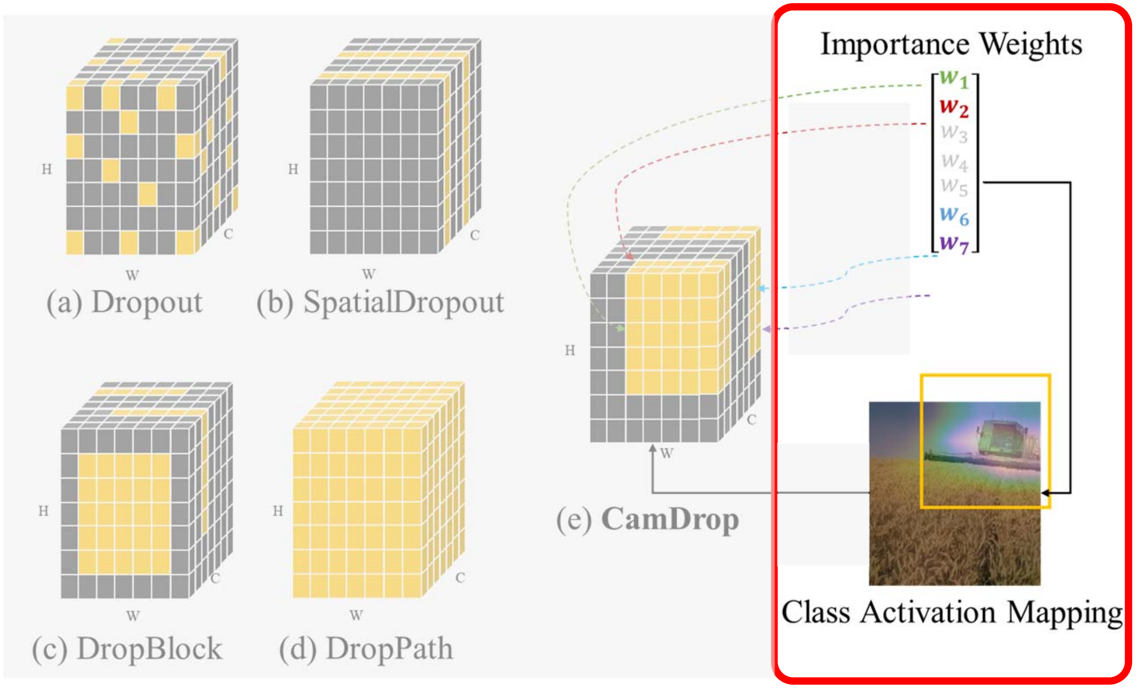
i. GAP layer:

$$J^{k'} = \sum_{c=1}^{C'} \alpha_c^{k'} Z_{c,:,:}$$

ii. Shallow Layers:

$$J^{k'} = \sum_{c=1}^{C''} \left(\sum_{j=1}^{\left\lfloor \frac{C'}{C''} \right\rfloor} \alpha_j^{k'} \right) X^l_{c,:,:}$$

where the weight vector $\alpha^{k'} \in \mathbb{R}^{C'\times 1}$indicates the importance of each channel for a specific class k' of y, Z, X are the output feature map of the penultimate/middle layer.

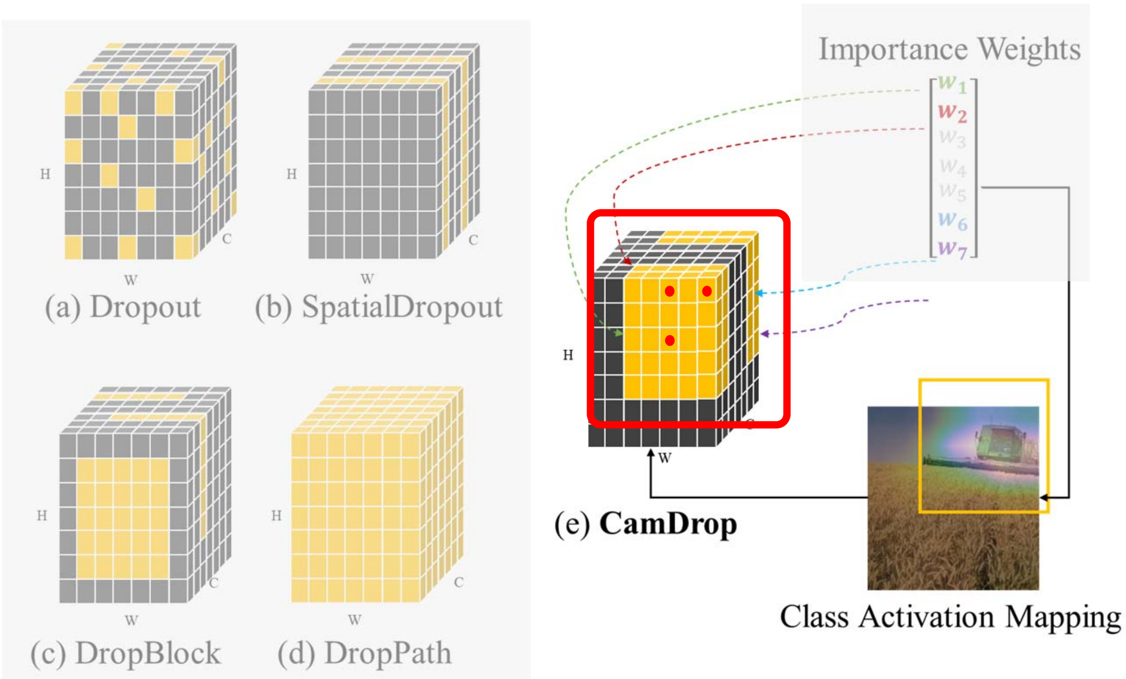


Method

(1) Spatial-wise binary mask $M^{(1)}$:

$$M_{h,w}^{(1)} = \begin{cases} 0, & J_{h,w}^{k'} > \inf\{T_n^s\} \\ 1, & J_{h,w}^{k'} < \inf\{T_n^s\} \end{cases}$$

$J^{k'}$ can be considered as a set with $H' \times W'$ elements $\{j_{1,1}, \ldots, j_{H' \times W'}\}$, each of which implies the significance of units at spatial grid (h, w).
T_n^s is the set of **the n most important pixels**.

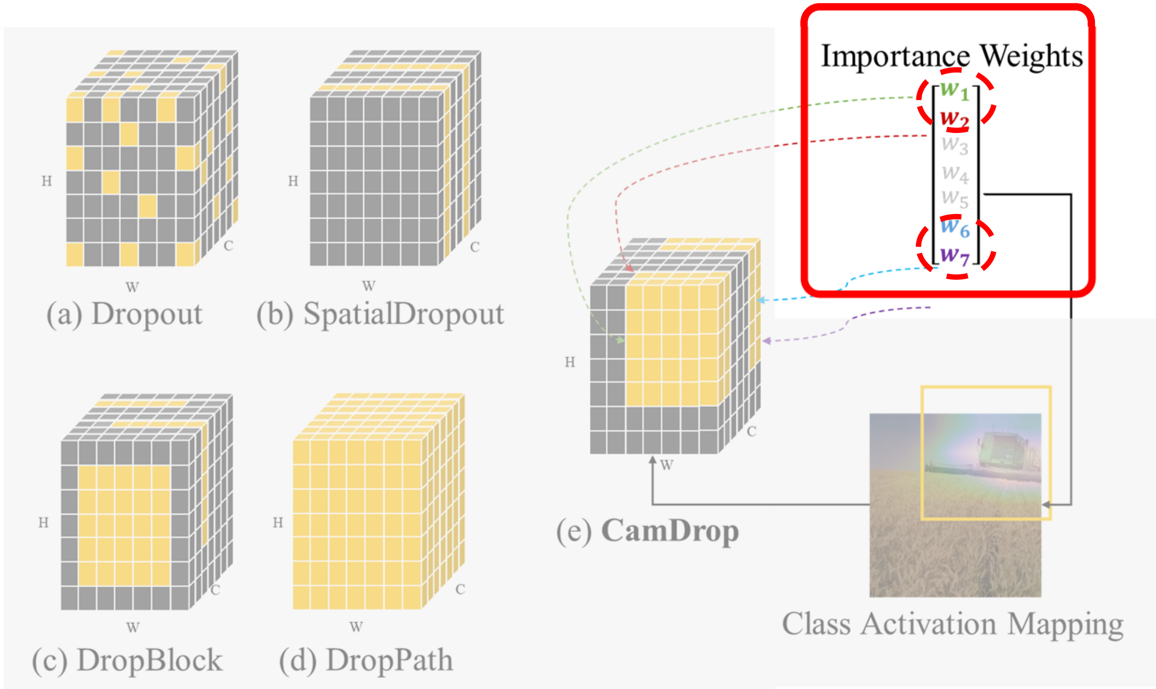


Method

(2) Depth-wise binary mask $M^{(2)}$:

$$M_c^{(2)} = \begin{cases} 0, & \alpha_c^{k'} > \inf\left\{T_{n'}^{d}\right\} \\ 1, & \alpha_c^{k'} < \inf\left\{T_{n'}^{d}\right\} \end{cases}$$

where n' is a hyperparameter and $\inf\{\cdot\}$ is the infimum of a set
$M_c^{(2)}$ is the set of **the n most important channels in $\alpha^{k'}$**



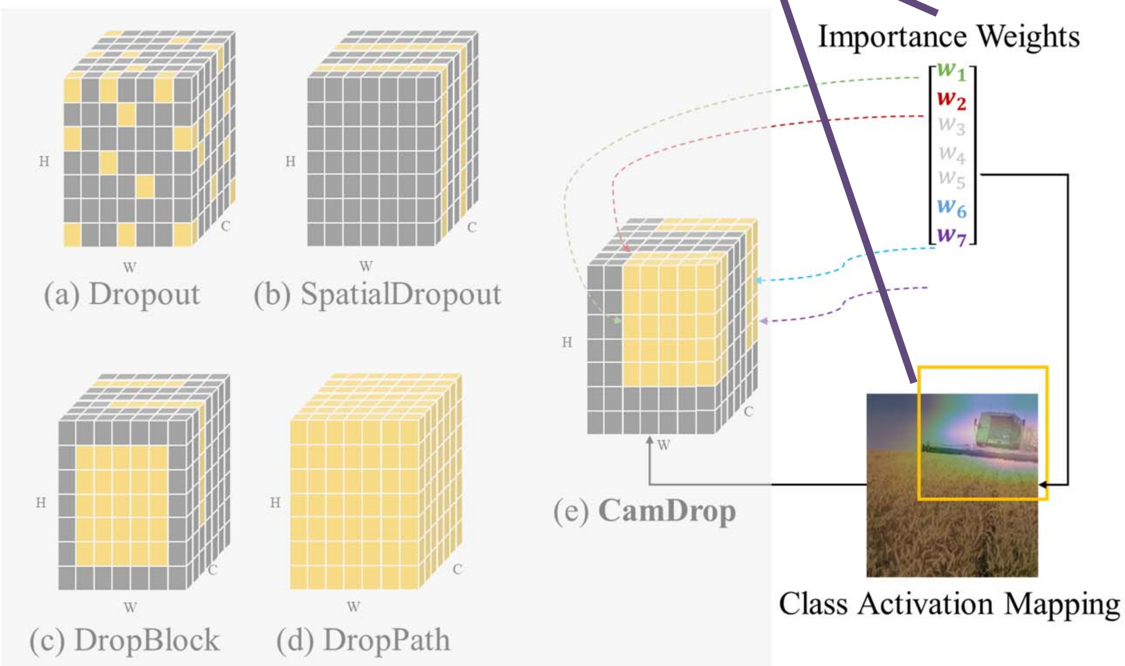
Method

(3) Let $M^{(3)}$ be the valid seed region of the feature map and ψ be a randomly sampled mask.

The combined mask $\boldsymbol{\mathcal{M}}$ is:

$$\mathcal{M} = \neg\psi \wedge \left(\bigwedge_i \neg M^{(i)}\right)$$

$\psi \sim Bernouli(\gamma)$

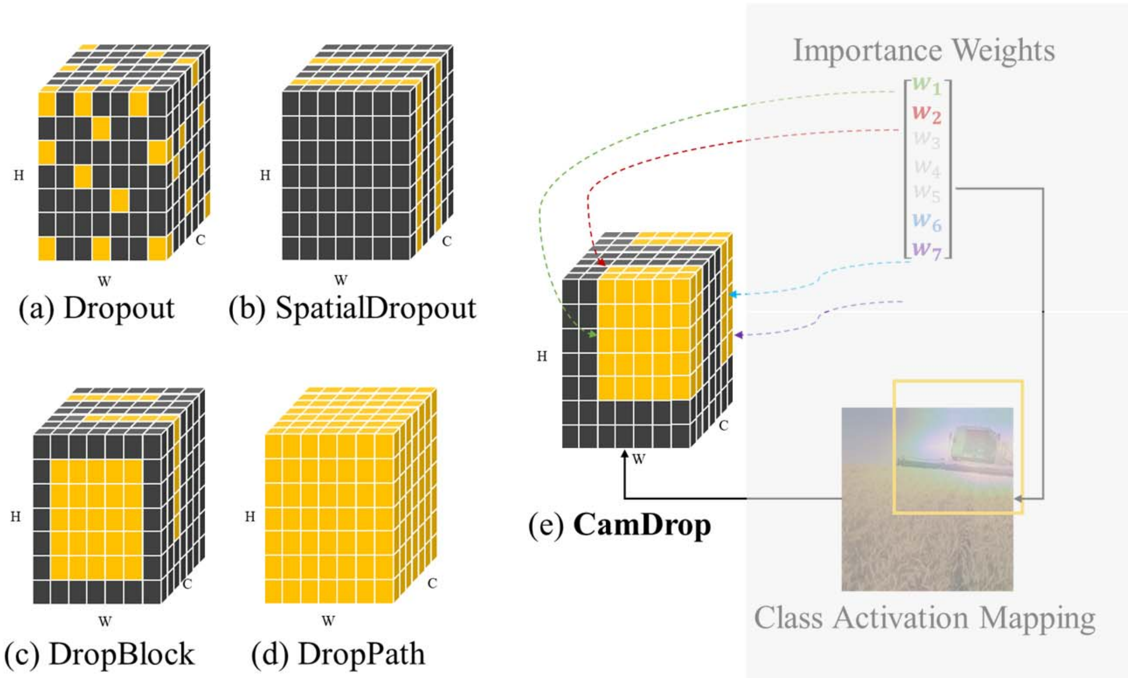


Method

(4) Sweep over $\boldsymbol{\mathcal{M}}$ and set pixels in $\{\forall q, \|q-u\|_1 = r\}$ zero, where u is the zero entries in $\boldsymbol{\mathcal{M}}$ and r is the length of mask.

We normalize $\boldsymbol{\mathcal{M}}$ by the factor:

$$\mathrm{M}^l = \frac{CHW}{\sum_{c=1,h=1,w=1}^{C,H,W} \mathcal{M}_{c,h,w}}$$



Outline

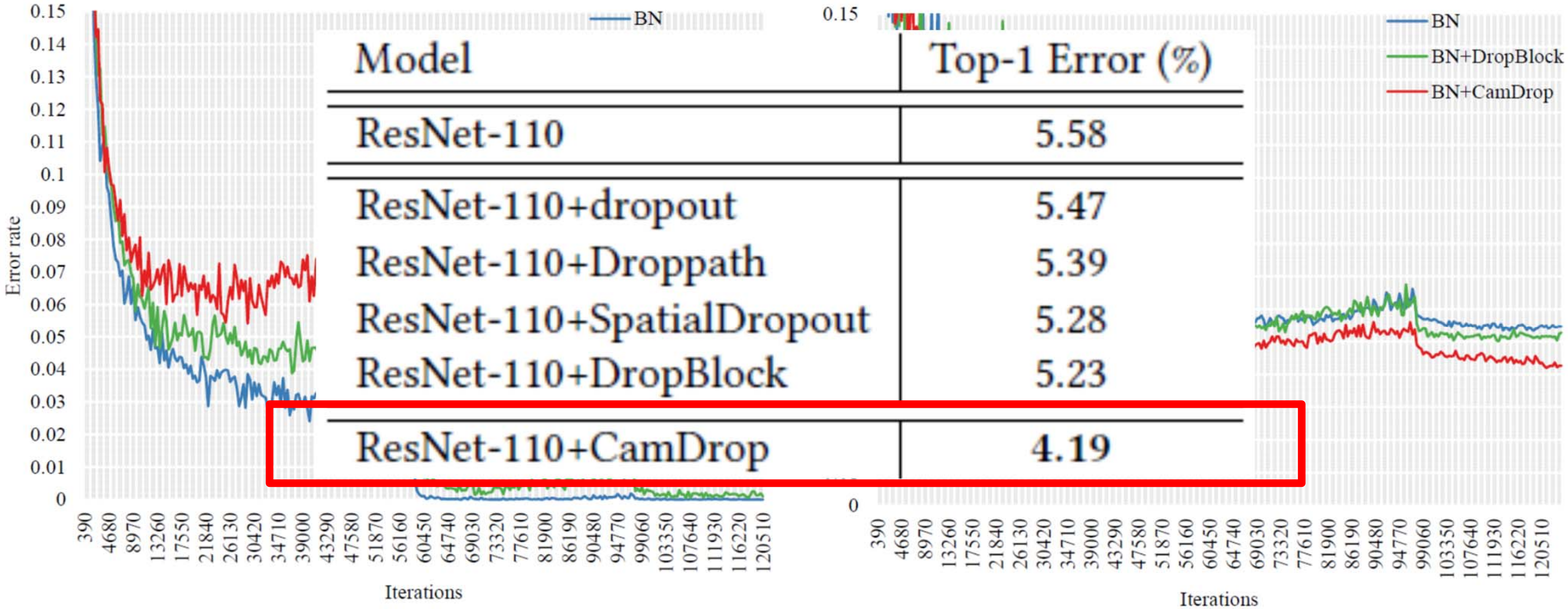
- Introduction
- Method
- **Experiment**
- Conclusion

Experiment

- Main Experiments
  - CIFAR10
  - ImageNet2012
- Ablation Studies
  - Effectiveness of CAM
  - Effectiveness of Importance Weights
  - Proportion of Dominant Visual Patterns
- More than Overfitting

Experiment

- Classification on CIFAR10

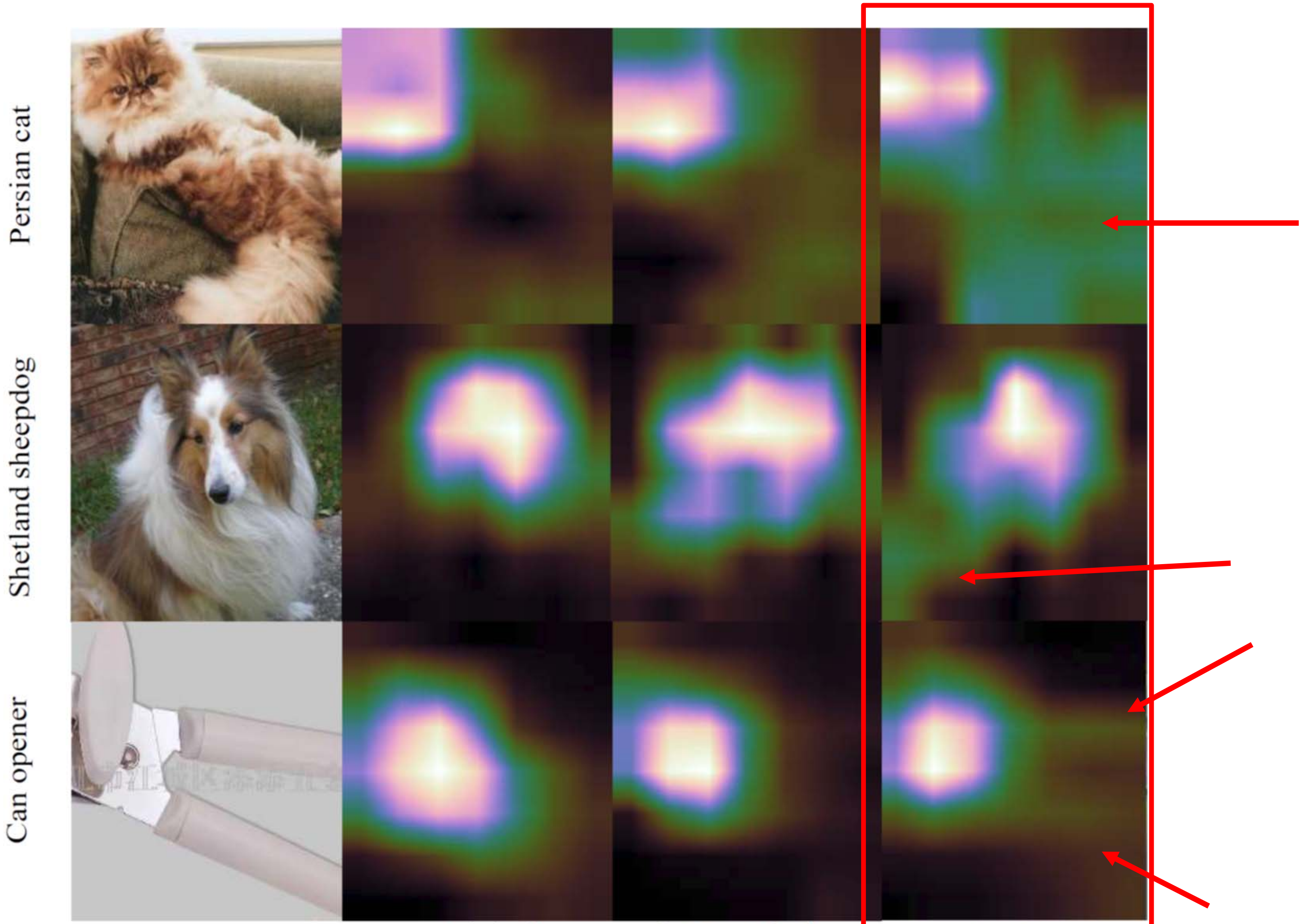


Model	Top-1 Error (%)
ResNet-110	5.58
ResNet-110+dropout	5.47
ResNet-110+Droppath	5.39
ResNet-110+SpatialDropout	5.28
ResNet-110+DropBlock	5.23
ResNet-110+CamDrop	4.19

(a) train error. (b) val error.

Experiment

- Classification on ImageNet

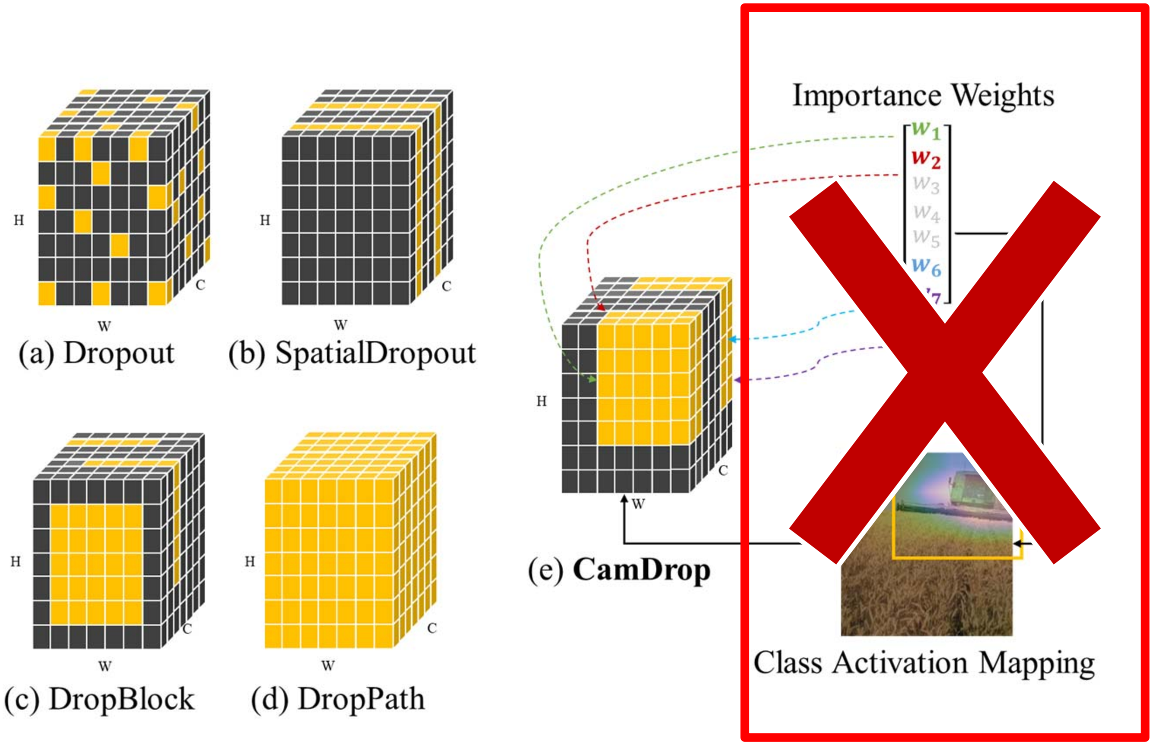


(a) Original (b) DropBlock (c) CamDrop

Ablation Studies

- Effectiveness of CAM

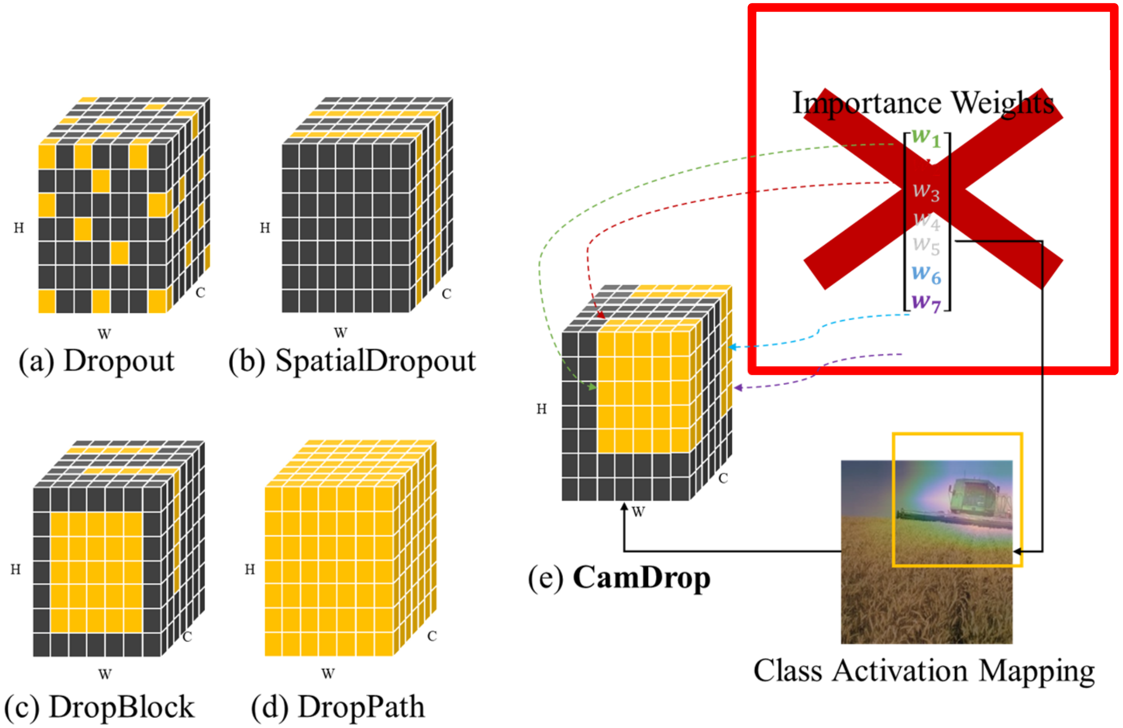
Average of feature map



	Top-1 Error (%)
Avg guided	5.07
CAM guided	**4.19**

Ablation Studies

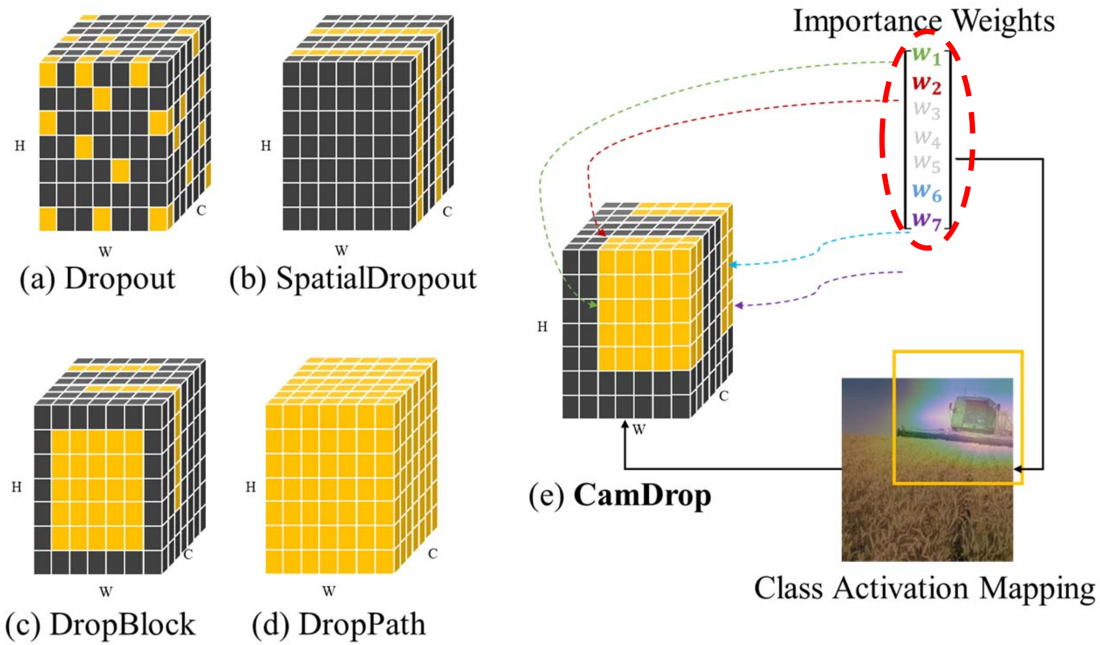
- Effectiveness of Importance Weights



	Top-1 Error (%)
All	4.51
Dominants	**4.19**

Ablation Studies

- Proportion of Dominant Visual Patterns



	Top-1 Error (%)
$n' = C/16$	4.39
$n' = C/8$	**4.19**
$n' = C/4$	4.75
$n' = C/2$	4.78

中山大學 SUN YAT-SEN UNIVERSITY

More than overfitting

- More than dealing with Overfitting
  - When inverting the decay scheme of learning rate, the validation errors of models trained with CamDrop are still close **even though the inversed one is overfitting** (**2nd/3rd rows**).
  - **The two overfitting with and without CamDrop** have a large gap (**1st/3rd rows**).

Model	Val Error (%)	Train Error (%)
ResNet-110, $lr = 2.0$	7.16	2.22e-4
ResNet-110, $lr = 2.0$ with CamDrop	5.94	8.53e-2
ResNet-110, $lr = 2.0$ with CamDrop, inversely	6.29	1.66e-4

More than overfitting

At the t-th update iteration in SGD, the weights and biases in the l-th layer will be updated as:

$$\mathbf{b}_t^l := \mathbf{b}_{t-1}^l - \eta \frac{\partial \mathcal{L}}{\partial \mathbf{b}_{t-1}^l}$$

According to the chain rules:

$$\frac{\partial \mathcal{L}}{\partial b_{t-1,i}^l} = \frac{\partial \mathcal{L}}{\partial \mathbf{S}} \frac{\partial \mathbf{S}}{\partial b_{t-1,i}^l}$$

The upper bounds of the loss can be given by the Hölder inequality:

$$\left| \frac{\partial \mathcal{L}}{\partial b_{t-1,i}^l} \right| \leq \max_{k'} \left| \frac{\partial \mathcal{L}}{\partial S_{k'}} \right| \left\| \frac{\partial \mathbf{S}}{\partial b_{t-1,i}^l} \right\|_1$$

More than overfitting

Since $|\partial \boldsymbol{L}/\partial \boldsymbol{S}| = |softmax(\boldsymbol{S}) - \boldsymbol{y}|$ cannot exceed 1 for any element, the inequality can be reduced to:

$$\left|\frac{\partial \mathcal{L}}{\partial b_{t-1,i}}\right| \leq \left\|\frac{\partial \mathbf{S}}{\partial b_{t-1,i}}\right\|_1$$

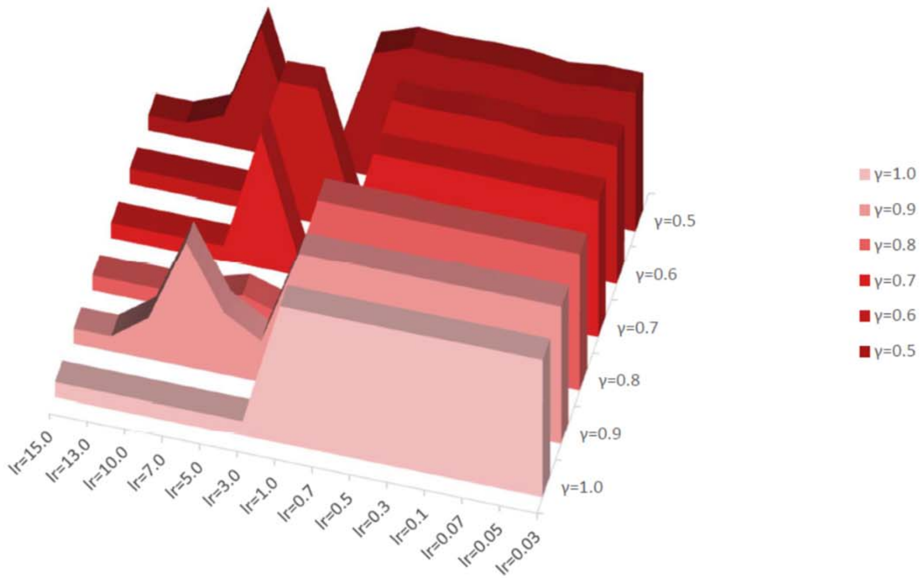
CamDrop masks out the several notable neurons, which gives a ***tighter upper bound*** of the update of biases $\partial L/\partial b$ at each iteration:

$$\left|\frac{\partial \mathcal{L}}{\partial b^l_{t-1,i}}\right| \leq \left\|\frac{\partial \mathbf{S}}{\partial b^l_{t-1,i}}\right\|_1^{(c)} \leq \left\|\frac{\partial \mathbf{S}}{\partial b^l_{t-1,i}}\right\|_1^{(t)} \leq \left\|\frac{\partial \mathbf{S}}{\partial b^l_{t-1,i}}\right\|_1$$

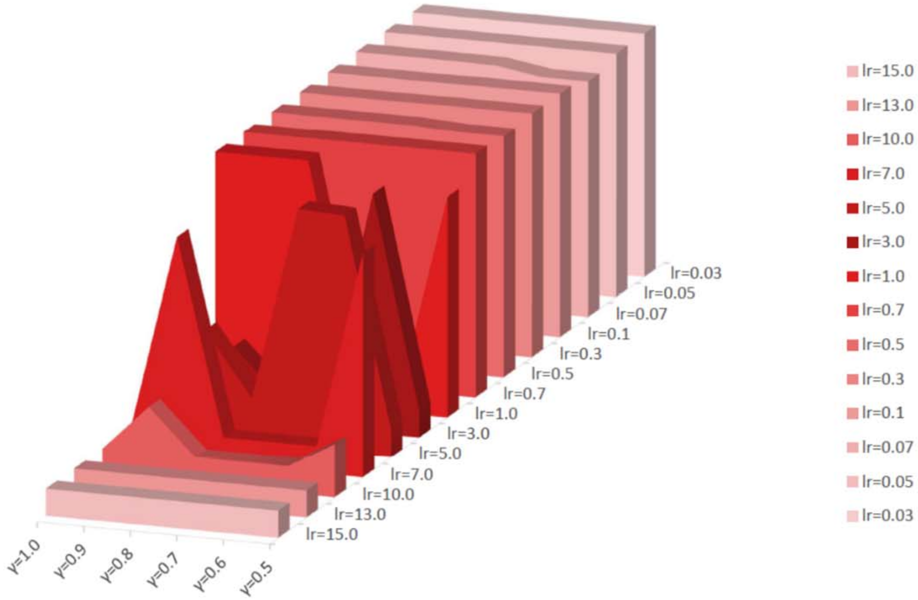
where $\|\cdot\|^{(c)}$ and $\|\cdot\|^{(t)}$ stand for the L_1-norm of derivative with Cam/traditional dropout mask respectively

More than overfitting

- Relationships among the learning rate, the remaining rate and top1-accuracy (z-axis)



(a)



(b)

Outline

- Introduction
- Method
- Experiment
- **Conclusion**

Conclusion

- Conclusion
  - A novel dropout method called **CamDrop** to improve the robustness of DNN models by **masking the dominant regions** with the guidance of **class activation mapping**.
  - Dropout techniques actually make **the upper bound of the magnitude of gradients much tighter**.
  - Data with **non-Euclidean structure** can utilize this technique by establishing graph with corresponding relationships between vertices and edges.

Thank you

Mail: wanghq8@mail2.sysu.edu.cn